On Delivering Embarrassingly Distributed Cloud Services

Ken Church
Albert Greenberg
James Hamilton
{church, albert, jamesrh}@microsoft.com

Board

$1B

Affordable

$2M

Containers:
Disruptive Technology

- Implications for Shipping
  - New Ships, Ports, Unions
- Implications for Hotnets
  - New Data Center Designs
  - Power/Networking Trade-offs
  - Cost Models: Expense vs. Capital
  - Apps: *Embarrassingly Distributed*
    - Restriction on *Embarrassingly Parallel*
  - Machine Models
    - Distributed Parallel Cluster $\subseteq$ Parallel Cluster

Related Work

- **http://en.wikipedia.org/wiki/Data_center**
  - A data center can occupy one room of a building…
  - Servers differ greatly in size from 1U servers to large … silos
  - Very large data centers may use shipping containers…[2]

220 containers in one PoP
→
220 in 220 PoPs

Mega vs. Micro Data Centers (DCs)

POPs	Cores/ POP	Hardware/ POP	Co-located With/Near
1	1,000,000	1000 containers	Mega Data Center
10	100,000	100 containers	
100	10,000	10 containers	Fiber Hotel/ Power Substation
1,000	1,000	1 container	
10,000	100	1 rack	Central Office
100,000	10	1 mini-tower	P2P
1,000,000	1	embedded	

Data Center (DC) Cost Models: Power vs. Networking

- Big ticket items: Contents, Power, Networking
  - Currently, cost of DC (excluding contents) ≈ cost of contents
    - Moore's Law: more appropriate for contents than DC
    - Eventually, cost of DC >> cost of contents
- Power: costs more than networking
  - Opportunity: Power
- Self-Serving Recommendation:
  - Save big $$ on power
  - By investing in what we do
    - Networking
    - Redesigning Apps (for geo-diversity)

Cost Models:
Expense vs. Capital

- Wide Agreement: Cost dominated by power
  - Less Agreement: Expense or Capital?
- Can we save $$ by putting clouds to sleep?
  - Yes, but only a little expense
- Big Opportunity: Capital
  - Capital: Batteries, Generators, Power Distribution
  - Worst case forecast (capital) >> Actual usage (expense)
- ottleneck: Sunk Costs (Independent of Usage)
  - Rights to consume power: like the last seat on an airplane
  - Putting clouds to sleep: Like stripping weight on last seat
    - Saves a little expense (fuel),
    - But better to sell last seat (even at a steep discount)

Variant on Buffet Principle ?

Use it or lose it (more applicable for sunk capital than expense)

Risk Management → Micro

- Risky and expensive
  - To put all our eggs in one basket (Mark Twain)
- Mega DC → Redundancy Mechanisms
  - Power: Batteries, Generators, Diesel Fuel
    - Over 20% of DC costs is in power redundancy
  - Networking: Protection (SONET)
- Micro Alternative: Geo-Redundancy
  - *N+1* Redundancy: More attractive for large *N*
  - Geo-Redundancy: Not appropriate for all apps
    - But many apps are *Embarrassingly Distributed*

Cost Model Recap

- Power costs dominate everything else
  - Save $$ on power
  - By investing in what we do
    - Networking
    - Re-designing apps (for geo-diversity)
- Capital dominates expense
  - Opportunity: Risk Management (Geo-Diversity)
    - *N*+1 Redundancy: More attractive for large *N*
    - Cheaper than Batteries & Generators

Economies of Scale: Mega/Micro Neutral

- Large cloud service providers
  - Amazon, Google, Microsoft, Yahoo, etc.
- Clouds enjoy economies of scale
  - Networking & Power:
    - Large Purchases → Favorable Terms
    - Though sometimes, Demand >> Supply → Unfavorable Terms
- Clouds pass savings on to consumers
- Myth: Economies of Scale → Mega
  - Large firms (Walmart) enjoy favorable terms because they are large
    - Independent of mega vs. micro (PoPs/Stores)
  - Economies of scale favor large volume (sales), not large PoPs

Micro is Better: Both Capital & Expense

		Mega (DC)	Micro (Condos)
Specs	Servers	**54k**	**54k** (= 48 servers/condo * 1125 Condos)
	Power (Peak)	**13.5 MW**	**13.5MW** (= 250 Watts/server * 54k servers = 12 KW/condo * 1125 Condos)

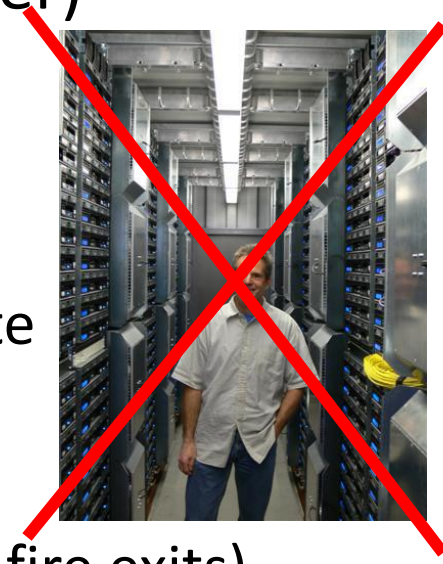
Clouds → Condos → Containers

- Whenever we see a crazy idea
  - Within 2x of current practice,
  - Something is wrong
- Let's go pick some low hanging fruit

- Though maybe not as crazy as we thought…
  - Changes are coming

Container Abstraction

- Modular Data Center (≈2k Servers/Container)
- Cheap Units: millions, not billions
  - Affordable, even by universities
- Just-in-Time:
  - Easy to build, provision, move, buy/sell, operate
- Sealed Boxcar: No Serviceable Parts
  - No Humans Inside
  - No room for people, too hot, noisy, unsafe (no fire exits)
  - Not OSHA Compliant
- Self-contained unit with everything but
  - Power: 480V
  - External Network: 1 Gbps
  - Cooling: Chilled Water



Containers are a Disruptive Technology: Implication for CS Theory/Algorithms

- Abstract Machine Model
  - Distributed Parallel Cluster $\subseteq$ Parallel Cluster
- Distributed Container Farm
  - Parallel Cluster (with Boundaries)
    - Better communication within containers
    - Than across containers (wide area network)
- Challenge:
  - Find appropriate apps (not for everything)
    - Apps that fit within boundaries
    - Or distribute nicely across them (Embarrassingly Distributed)
- Embarrassingly Distributed:
  - Stronger Condition than Embarrassingly Parallel
  - Map Reduce, Sort, Scatter Gather

Embarrassingly Distributed Apps

- Currently distributed:
  - voice mail, telephony (Skype), P2P file sharing (Napster), multicast, eBay, online games (Xbox Live), grid computing
- Obvious candidates:
  - spam filtering & email (Hotmail), backup, grep (simple but common forms of searching through a large corpus)
- Less obvious candidates:
  - map reduce computations (in the most general case), sort (in the most general case), social networking (Facebook)

Obvious Candidate: Email on the Edge

- Hotmail: Four Activities
  - Incoming Email Anti-malware and Routing
  - Email Storage Management
  - Users and Administrator Service
  - Outgoing Email Service
- All are embarrassingly distributed
  - Useful Option: Distribute some and centralize others
  - Routing/Load-Balancing/Blocking: Deploy near user
    - Edge Blocks (email) & Call Gapping (telephony)
      - If traffic will be blocked,
      - Better to deploy blocks near source (to save transport)

Networking

- DC→ WAN → Peering → Last Mile
- Mega — Build
  - Build global dedicated backbone between DCs with rich peering (and plenty of redundancy/reliability)
  - Relatively complex, high cost, but high control of quality
- Micro — Buy
  - Buy transit from network service providers (CDN)
  - Vastly simpler and lower cost, but with reduced control of quality; quality via redundancy and global load balancing
  - Closer to user: latency, edge blocking options

Fragmentation Tax
(For Embarrassingly Distributed Apps)

- Split a mega data center into *K* micro data centers
  - Redundancy Tax: Unit cost ($U \geq 1$) = Capital in mega/micro
  - Fragmentation Tax (Capital)
- With global load balancing: No Fragmentation Tax
- Without global load balancing, … it depends:

$$m(U-1) - n\sqrt{ma}(\sqrt{K}-U)$$

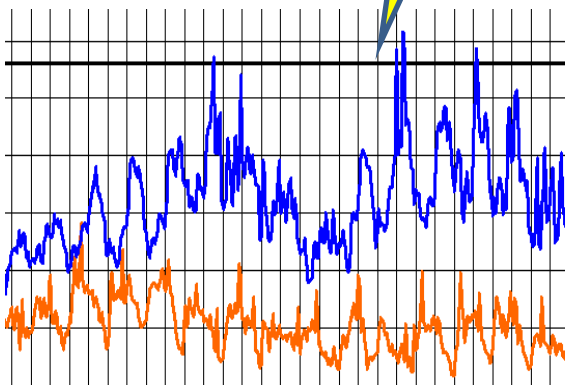
Mean traffic rate

Unit cost ≥ 1
Capital: Mega/Micro
Redundancy Tax

SLA

Traffic Burstiness

SLA



  - $U > 1$: Unit costs dominate (for large m) → Micro cheaper
  - U=1: Mega cheaper with fragmentation tax scaling slowly

Redundancy tax dominates fragmentation tax

$1B

Board

Conclusions

Current Status: Long on Mega

$2M

Affordable

is aggressively adopting

- Industry is investing billions in mega
  - Lots of new mega Data Centers
  - Long-Term Assets/Liabilities
    - Depreciating over 15 years
- Bottom line:
  - The industry ~~would do well to consider~~
  - The micro alternative
    - Geo-Diversity >> Batteries & Generators
      - Fragmentation Tax (sublinear) << Redundancy Tax (linear)
    - Just-in-Time Options:
      - Easy to build, provision, move, buy/sell, operate
    - Risk (hedge the long position on mega)

BACKUP

3 >> 10 ???

- Cost Drivers:
  - Market Segmentation: Business vs. Consumer
    - Businesses pay more because of willingness to pay
    - Like hotels, airplanes, telephones, etc.
    - 3 cents/KW >> 10 cents/KW
  - Capital (not Expense): cents >> 10 cents (???)
    - Costs are dominated by backup systems, transformers, custom power distribution networks, etc.
- Opportunities:
  - Just-in-Time options to ramp up/down investments quickly

Large Data Centers (DCs): Analogous to Large Conferences

- A small (low budget) workshop can be held in a spare room in many universities,
  - but costs escalate rapidly for larger meetings that require hotels and convention centers.
- There are thousands of places where the current infrastructure can accommodate a workshop or two,
  - But no place where the current infrastructure can accommodate the Olympics
    - Without a significant capital investment.
- Meetings encounter diseconomies of scale when they outgrow the capabilities of off-the-shelf venues.

So too, costs escalate for mega DC

- For example, if mega data center (DC) consumes 20MW of power at peak from a given power grid,
  - that grid may be unable or unwilling to sell another 20MW to the same data center operator.
- In general, new mega DC → significant capital investments
  - Building, power, networking
- Micro alternative → More opportunity for reuse
  - Exploit overbuild in power grid and networking fabric.
- Many places can handle a container (power substations)
  - but no place for lots of containers (without significant investment)
- Data centers encounter diseconomies of scale
  - When they become so large that they require
  - Significant investment in infrastructure.

Just-In-Time

- Micro: options to buy/sell just-in-time
- Smooth investments over time:
  - Mega ($1B/year) → Micro ($20M/week)
- Lead times on new orders:
  - Mega DC: 1 year
  - Micro: Container production line (approx 1/week)
- Options to unwind investments
  - Micro >> Mega (15 year depreciation)

Just-in-Time Options: Valuable under Uncertainty

- Long-term demand is far from flat and certain
  - Demand for cloud services will probably increase,
  - But anything could happen over next 15 years
- Short-term demand is far from flat and certain
  - Power usage depends on many factors including time of day, day of week, seasonality, economic booms and busts